

# 모두의 핏 앱 API 명세서 v2

## 1. 문서 개요

### 1.1 프로젝트 정보

- **프로젝트명:** 모두의 핏 (All4Fit) 앱
- **API 버전:** v2
- **Base URL:** <https://api.all4fit.co.kr/v2>
- **인증 방식:** JWT Bearer Token (roles 포함)
- **응답 형식:** JSON
- **작성일:** 2025년 1월

### 1.2 공통 응답 형식

#### 성공 응답

```
{  
    "success": true,  
    "message": "요청이 성공적으로 처리되었습니다.",  
    "data": {  
        // 응답 데이터  
    }  
}
```

#### 에러 응답

```
{  
    "success": false,  
    "message": "에러 메시지",  
    "error": {  
        "code": "ERROR_CODE",  
        "details": "상세 에러 정보"  
    }  
}
```

### 1.3 HTTP 상태 코드

- **200:** 성공
- **201:** 생성 성공
- **400:** 잘못된 요청
- **401:** 인증 실패
- **403:** 권한 없음
- **404:** 리소스 없음

- 409: 충돌 (중복 등)
- 500: 서버 오류

## 1.4 JWT 토큰 구조

```
{  
  "userId": 123,  
  "email": "user@example.com",  
  "roles": ["1", "2"],  
  "iat": 1234567890,  
  "exp": 1234571490  
}
```

### roles 설명:

- 1: 일반 사용자
- 2: 지도자
- 3: 시설 운영자
- 4: 관리자

## 2. 인증 API

### 2.1 통합 로그인 (LOGIN\_UNIFIED)

#### POST /auth/login

이메일/전화번호와 비밀번호로 로그인합니다.

#### 요청 헤더:

```
Content-Type: application/json
```

#### 요청 본문:

```
{  
  "loginId": "user@example.com",  
  "password": "password123!"  
}
```

#### 요청 매개변수:

필드명	타입	필수	설명
loginId	string	O	이메일 또는 전화번호
password	string	O	비밀번호

**응답 본문:**

```
{  
    "success": true,  
    "message": "로그인이 성공했습니다.",  
    "data": {  
        "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",  
        "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",  
        "expiresIn": 3600,  
        "user": {  
            "userId": 123,  
            "email": "user@example.com",  
            "nickname": "길동이",  
            "roles": ["1", "2"],  
            "profileImage": "https://example.com/profile.jpg"  
        }  
    }  
}
```

## 2.2 일반 사용자 회원가입 (SIGNUP)

**POST /auth/register**

일반 사용자 회원가입을 처리합니다.

**요청 본문:**

```
{  
    "email": "user@example.com",  
    "phone": "010-1234-5678",  
    "password": "password123!",  
    "name": "홍길동",  
    "nickname": "길동이",  
    "birthDate": "1990-01-01",  
    "gender": "M",  
    "address": "서울시 강남구 테헤란로 123",  
    "termsAgreed": {  
        "service": true,  
        "privacy": true,  
        "marketing": false,  
        "location": true  
    }  
}
```

**응답 본문:**

```
{  
    "success": true,
```

```
"message": "회원가입이 완료되었습니다.",  
"data": {  
    "userId": 123,  
    "email": "user@example.com",  
    "nickname": "길동이",  
    "roles": ["1"],  
    "status": "ACTIVE",  
    "createdAt": "2024-10-01T00:00:00Z"  
}  
}
```

## 2.3 파트너 등록 - 기본 정보 (PARTNER\_REGISTRATION)

### POST /auth/partner/register/basic

파트너 등록의 기본 정보(일반 사용자 가입)를 처리합니다.

#### 요청 본문:

```
{  
    "email": "partner@example.com",  
    "phone": "010-1234-5678",  
    "password": "password123!",  
    "name": "김파트너",  
    "nickname": "김파트너",  
    "birthDate": "1985-01-01",  
    "gender": "M",  
    "address": "서울시 강남구 테헤란로 123",  
    "termsAgreed": {  
        "service": true,  
        "privacy": true,  
        "marketing": false,  
        "location": true  
    }  
}
```

#### 응답 본문:

```
{  
    "success": true,  
    "message": "파트너 기본 정보가 등록되었습니다.",  
    "data": {  
        "userId": 123,  
        "email": "partner@example.com",  
        "nickname": "김파트너",  
        "roles": ["1"],  
        "status": "ACTIVE",  
        "registrationToken": "temp_token_for_partner_registration",  
        "createdAt": "2024-10-01T00:00:00Z"  
    }  
}
```

```
    }  
}
```

## 2.4 파트너 탑입 선택 - 지도자 등록

### POST /auth/partner/register/instructor

지도자 파트너로 등록합니다.

#### 요청 헤더:

```
Authorization: Bearer temp_token_for_partner_registration  
Content-Type: multipart/form-data
```

#### 요청 본문:

```
{  
    "businessNumber": "123-45-67890",  
    "certification": "생활체육지도사 2급",  
    "experience": "10년간 헬스장에서 근무",  
    "specializations": ["헬스", "필라테스"],  
    "bio": "건강한 삶을 위한 운동 지도자입니다.",  
    "hourlyRate": 50000,  
    "snsAccounts": {  
        "instagram": "@fitness_coach",  
        "youtube": "fitness_channel"  
    },  
    "facilityInfo": {  
        "facilityName": "강남 헬스장",  
        "facilityAddress": "서울시 강남구 테헤란로 123",  
        "postalCode": "06292",  
        "latitude": 37.5665,  
        "longitude": 126.9780  
    }  
}
```

#### 파일 필드:

- `businessLicense`: 사업자 등록증 (이미지)
- `certificationFile`: 자격증 (이미지)
- `profileImage`: 프로필 이미지
- `portfolioImages[]`: 활동 사진들 (최대 10개)

#### 응답 본문:

```
{  
    "success": true,
```

```
"message": "지도자 등록이 완료되었습니다. 승인 후 이용 가능합니다.",  
"data": {  
    "userId": 123,  
    "instructorId": 456,  
    "roles": ["1", "2"],  
    "status": "PENDING",  
    "message": "관리자 승인 후 지도자 권한이 활성화됩니다."  
}  
}
```

## 2.5 파트너 탑입 선택 - 시설 운영자 등록

### POST /auth/partner/register/facility-operator

시설 운영자 파트너로 등록합니다.

#### 요청 헤더:

```
Authorization: Bearer temp_token_for_partner_registration  
Content-Type: multipart/form-data
```

#### 요청 본문:

```
{  
    "representativeName": "김대표",  
    "businessNumber": "123-45-67890",  
    "facilityInfo": {  
        "name": "강남 헬스장",  
        "address": "서울시 강남구 테헤란로 123",  
        "postalCode": "06292",  
        "latitude": 37.5665,  
        "longitude": 126.9780,  
        "phone": "02-1234-5678",  
        "facilityType": "GYM",  
        "sports": [1, 2, 3]  
    },  
    "operatingHours": {  
        "monday": "06:00-24:00",  
        "tuesday": "06:00-24:00",  
        "wednesday": "06:00-24:00",  
        "thursday": "06:00-24:00",  
        "friday": "06:00-24:00",  
        "saturday": "08:00-22:00",  
        "sunday": "08:00-22:00"  
    },  
    "holidays": "매월 둘째, 넷째 일요일"  
}
```

## 파일 필드:

- `businessLicense`: 사업자 등록증 (이미지)
- `facilityImages[ ]`: 시설 사진들 (최대 20개)

## 응답 본문:

```
{
  "success": true,
  "message": "시설 운영자 등록이 완료되었습니다. 승인 후 이용 가능합니다.",
  "data": {
    "userId": 123,
    "operatorId": 789,
    "roles": ["1", "3"],
    "status": "PENDING",
    "message": "관리자 승인 후 시설 운영자 권한이 활성화됩니다."
  }
}
```

## 2.6 파트너 탑입 선택 - 통합 등록

### POST /auth/partner/register/integrated

지도자 + 시설 운영자로 통합 등록합니다.

#### 요청 헤더:

```
Authorization: Bearer temp_token_for_partner_registration
Content-Type: multipart/form-data
```

#### 요청 본문:

```
{
  "instructor": {
    "businessNumber": "123-45-67890",
    "certification": "생활체육지도사 2급",
    "experience": "10년간 헬스장에서 근무",
    "specializations": ["헬스", "필라테스"],
    "bio": "건강한 삶을 위한 운동 지도자입니다.",
    "hourlyRate": 50000,
    "snsAccounts": {
      "instagram": "@fitness_coach"
    }
  },
  "facility": {
    "representativeName": "김대표",
    "businessNumber": "987-65-43210",
    "name": "강남 헬스장",
  }
}
```

```
        "address": "서울시 강남구 테헤란로 123",
        "postalCode": "06292",
        "latitude": 37.5665,
        "longitude": 126.9780,
        "phone": "02-1234-5678",
        "facilityType": "GYM",
        "sports": [1, 2, 3],
        "operatingHours": {
            "monday": "06:00-24:00",
            "tuesday": "06:00-24:00",
            "wednesday": "06:00-24:00",
            "thursday": "06:00-24:00",
            "friday": "06:00-24:00",
            "saturday": "08:00-22:00",
            "sunday": "08:00-22:00"
        },
        "holidays": "매월 둘째, 넷째 일요일"
    }
}
```

#### 파일 필드:

- instructorBusinessLicense: 지도자 사업자 등록증
- instructorCertification: 자격증
- instructorProfileImage: 프로필 이미지
- instructorPortfolio[]: 활동 사진들
- facilityBusinessLicense: 시설 사업자 등록증
- facilityImages[]: 시설 사진들

#### 응답 본문:

```
{
    "success": true,
    "message": "통합 파트너 등록이 완료되었습니다. 승인 후 이용 가능합니다.",
    "data": {
        "userId": 123,
        "instructorId": 456,
        "operatorId": 789,
        "roles": ["1", "2", "3"],
        "status": "PENDING",
        "message": "관리자 승인 후 지도자 및 시설 운영자 권한이 활성화됩니다."
    }
}
```

## 2.7 토큰 갱신

### POST /auth/refresh

액세스 토큰을 갱신합니다.

## 요청 헤더:

```
Authorization: Bearer refresh_token
```

## 응답 본문:

```
{
  "success": true,
  "message": "токен이 갱신되었습니다.",
  "data": {
    "accessToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "refreshToken": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...",
    "expiresIn": 3600
  }
}
```

## 2.8 중복 확인 API

### POST /auth/check-email

이메일 중복 확인을 합니다.

## 요청 본문:

```
{
  "email": "user@example.com"
}
```

## 응답 본문:

```
{
  "success": true,
  "data": {
    "available": true,
    "message": "사용 가능한 이메일입니다."
  }
}
```

### POST /auth/check-phone

전화번호 중복 확인을 합니다.

### POST /auth/check-nickname

닉네임 중복 확인을 합니다.

## 2.9 전화번호 인증 API

### POST /auth/send-verification

SMS 인증번호를 발송합니다.

요청 본문:

```
{  
  "phone": "010-1234-5678",  
  "type": "SIGNUP"  
}
```

## 3. 사용자 관리 API

### 3.1 프로필 조회

#### GET /users/profile

현재 사용자의 프로필 정보를 조회합니다.

요청 헤더:

```
Authorization: Bearer access_token
```

응답 본문:

```
{  
  "success": true,  
  "data": {  
    "userId": 123,  
    "email": "user@example.com",  

```

### 3.2 권한 정보 조회

#### GET /users/roles

현재 사용자의 권한 정보를 조회합니다.

요청 헤더:

```
Authorization: Bearer access_token
```

응답 본문:

```
{
  "success": true,
  "data": {
    "userId": 123,
    "roles": ["1", "2", "3"],
    "roleDetails": {
      "user": {
        "role": "1",
        "status": "ACTIVE",
        "registeredAt": "2024-10-01T00:00:00Z"
      },
      "instructor": {
        "role": "2",
        "instructorId": 456,
        "status": "APPROVED",
        "approvedAt": "2024-10-02T00:00:00Z"
      },
      "facilityOperator": {
        "role": "3",
        "operatorId": 789,
        "status": "PENDING",
        "appliedAt": "2024-10-03T00:00:00Z"
      }
    }
  }
}
```

### 3.3 권한 신청

#### POST /users/roles/apply

새로운 권한을 신청합니다.

요청 헤더:

```
Authorization: Bearer access_token  
Content-Type: multipart/form-data
```

### 요청 본문:

```
{  
    "roleType": "INSTRUCTOR",  
    "businessNumber": "123-45-67890",  
    "certification": "생활체육지도사 2급",  
    "experience": "10년간 헬스장에서 근무",  
    "specializations": ["헬스", "필라테스"]  
}
```

### 파일 필드:

- **businessLicense**: 사업자 등록증
- **certificationFile**: 자격증

### 응답 본문:

```
{  
    "success": true,  
    "message": "지도자 권한 신청이 완료되었습니다.",  
    "data": {  
        "applicationId": 999,  
        "roleType": "INSTRUCTOR",  
        "status": "PENDING",  
        "appliedAt": "2024-10-01T00:00:00Z"  
    }  
}
```

## 4. 시설 관련 API

### 4.1 시설 목록 조회 (FACILITY\_LIST)

#### GET /facilities

시설 목록을 조회합니다.

#### 쿼리 매개변수:

매개변수	타입	필수	기본값	설명
page	integer	X	1	페이지 번호
limit	integer	X	20	페이지당 항목 수

매개변수	타입	필수	기본값	설명
keyword	string	X		검색 키워드
facilityType	string	X		시설 유형 (GYM, POOL, TENNIS, etc.)
sportId	integer	X		운동 종목 ID
latitude	float	X		사용자 위도
longitude	float	X		사용자 경도
radius	integer	X	5000	검색 반경 (미터)
sortBy	string	X	distance	정렬 기준 (distance, rating, price)
sortOrder	string	X	asc	정렬 순서 (asc, desc)

## 5. 지도 관련 API

### 5.1 지도 시설 조회 (MAP\_TAB)

#### GET /map/facilities

지도 영역 내의 시설을 조회합니다.

### 5.2 자기 위치 이동 (MAP\_MY\_LOCATION) - NEW

#### GET /map/my-location

사용자의 현재 위치를 조회합니다.

요청 헤더:

```
Authorization: Bearer access_token
```

쿼리 매개변수:

매개변수	타입	필수	설명
latitude	float	O	현재 위도
longitude	float	O	현재 경도

응답 본문:

```
{
  "success": true,
  "data": {
    "latitude": 37.5665,
    "longitude": 126.9780,
    "address": "서울특별시 중구 태평로1가",
  }
}
```

```
"nearbyFacilities": [
    {
        "facilityId": 1,
        "name": "강남 헬스장",
        "distance": 150,
        "latitude": 37.5670,
        "longitude": 126.9785
    }
]
```

## POST /map/update-location

사용자의 위치를 업데이트합니다.

### 요청 헤더:

```
Authorization: Bearer access_token
Content-Type: application/json
```

### 요청 본문:

```
{
    "latitude": 37.5665,
    "longitude": 126.9780,
    "address": "서울특별시 중구 태평로1가"
}
```

### 응답 본문:

```
{
    "success": true,
    "message": "위치가 업데이트되었습니다.",
    "data": {
        "locationId": 1,
        "latitude": 37.5665,
        "longitude": 126.9780,
        "address": "서울특별시 중구 태평로1가",
        "updatedAt": "2024-10-01T12:00:00Z"
    }
}
```

## 6. 관리자 API

## 6.1 권한 승인 관리

### GET /admin/applications

권한 신청 목록을 조회합니다.

요청 헤더:

```
Authorization: Bearer access_token (role = 4 필요)
```

쿼리 매개변수:

매개변수	타입	필수	기본값	설명
roleType	string	X		권한 유형 (INSTRUCTOR, OPERATOR)
status	string	X	PENDING	상태 (PENDING, APPROVED, REJECTED)
page	integer	X	1	페이지 번호
limit	integer	X	20	페이지당 항목 수

응답 본문:

```
{
  "success": true,
  "data": [
    {
      "userId": 123,
      "roleType": "INSTRUCTOR",
      "instructorId": 456,
      "userName": "김지도자",
      "email": "instructor@example.com",
      "businessNumber": "123-45-67890",
      "status": "PENDING",
      "appliedAt": "2024-10-01T00:00:00Z",
      "documents": {
        "businessLicense": "https://example.com/docs/license.jpg",
        "certification": "https://example.com/docs/cert.jpg"
      }
    }
  ],
  "pagination": {
    "page": 1,
    "limit": 20,
    "total": 50,
    "totalPages": 3
  }
}
```

## POST /admin/applications/{applicationId}/approve

권한 신청을 승인합니다.

요청 헤더:

```
Authorization: Bearer access_token (role = 4 필요)
```

응답 본문:

```
{
  "success": true,
  "message": "지도자 권한이 승인되었습니다.",
  "data": {
    "userId": 123,
    "roleType": "INSTRUCTOR",
    "status": "APPROVED",
    "approvedAt": "2024-10-01T12:00:00Z",
    "updatedRoles": ["1", "2"]
  }
}
```

## POST /admin/applications/{applicationId}/reject

권한 신청을 거부합니다.

요청 본문:

```
{
  "reason": "제출 서류 불충분"
}
```

## 7. 에러 코드

### 7.1 인증 관련 에러

코드	HTTP 상태	설명
AUTH_001	401	인증 토큰이 유효하지 않습니다
AUTH_002	401	토큰이 만료되었습니다
AUTH_003	403	권한이 없습니다
AUTH_004	409	이미 존재하는 이메일입니다
AUTH_005	409	이미 존재하는 전화번호입니다

코드	HTTP 상태	설명
AUTH_006	409	이미 존재하는 닉네임입니다
AUTH_007	403	해당 권한이 없습니다
AUTH_008	409	이미 신청 중인 권한입니다

## 7.2 사용자 관련 에러

코드	HTTP 상태	설명
USER_001	404	사용자를 찾을 수 없습니다
USER_002	400	잘못된 비밀번호입니다
USER_003	400	입력 데이터가 유효하지 않습니다
USER_004	403	승인 대기 중인 권한입니다

## 7.3 시설 관련 에러

코드	HTTP 상태	설명
FACILITY_001	404	시설을 찾을 수 없습니다
FACILITY_002	403	시설 등록 권한이 없습니다
FACILITY_003	400	잘못된 시설 정보입니다

# 8. API 사용 예시

## 8.1 파트너 등록 플로우

```
// 1. 기본 정보 등록 (일반 사용자 가입)
const basicResponse = await fetch('/api/v2/auth/partner/register/basic', {
  method: 'POST',
  headers: { 'Content-Type': 'application/json' },
  body: JSON.stringify({
    email: 'partner@example.com',
    phone: '010-1234-5678',
    password: 'password123!',
    name: '김파트너',
    nickname: '김파트너',
    birthDate: '1985-01-01',
    gender: 'M',
    address: '서울시 강남구',
    termsAgreed: {
      service: true,
      privacy: true,
      marketing: false,
      location: true
    }
  })
})
```

```

    })
});

const { registrationToken } = await basicResponse.json();

// 2. 파트너 탑입 선택 - 지도자 등록
const formData = new FormData();
formData.append('businessNumber', '123-45-67890');
formData.append('certification', '생활체육지도사 2급');
formData.append('businessLicense', businessLicenseFile);
formData.append('certificationFile', certificationFile);

const instructorResponse = await fetch('/api/v2/auth/partner/register/instructor',
{
  method: 'POST',
  headers: {
    'Authorization': `Bearer ${registrationToken}`
  },
  body: formData
});

const result = await instructorResponse.json();
// 결과: roles = ["1", "2"], status = "PENDING"

```

## 8.2 권한 기반 접근 제어

```

// JWT 토큰에서 roles 확인
const token = 'eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...';
const decoded = jwt.decode(token);

// roles: ["1", "2", "3"] - 일반사용자 + 지도자 + 시설운영자

// 권한 확인
if (decoded.roles.includes('2')) {
  // 지도자 전용 기능 접근 가능
}

if (decoded.roles.includes('3')) {
  // 시설 운영자 전용 기능 접근 가능
}

```

## 8.3 지도 자기 위치 이동

```

// 현재 위치 조회
const locationResponse = await fetch(`/api/v2/map/my-location?
latitude=37.5665&longitude=126.9780`, {
  headers: {
    'Authorization': `Bearer ${accessToken}`
  }
}

```

```
});  
  
const locationData = await locationResponse.json();  
// 주변 시설 정보와 함께 현재 위치 반환  
  
// 위치 업데이트  
await fetch('/api/v2/map/update-location', {  
  method: 'POST',  
  headers: {  
    'Authorization': `Bearer ${accessToken}`,  
    'Content-Type': 'application/json'  
  },  
  body: JSON.stringify({  
    latitude: 37.5665,  
    longitude: 126.9780,  
    address: '서울특별시 종구'  
  })
});
```

## 9. 주요 변경사항

### 9.1 역할 기반 권한 시스템

- **기존:** userType 단일 필드
- **변경:** roles 배열로 다중 권한 지원
- JWT 토큰에 roles 포함

### 9.2 파트너 등록 통합 API

- **기존:** 별도의 지도자/시설운영자 회원가입 API
- **변경:** 통합 파트너 등록 플로우 API
- 단계별 API 분리 (기본정보 → 파트너타입선택)

### 9.3 권한 관리 API 추가

- 권한 정보 조회 API
- 권한 신청 API
- 관리자 승인/거부 API

### 9.4 지도 기능 강화

- 자기 위치 조회 API
- 위치 업데이트 API
- 주변 시설 정보 포함

---

작성일: 2025년 1월

버전: 2.0

작성자: 김진석

기반 문서: 1차 개발 기능 정의서 v2, DB 설계서 v2

개발 범위: 1차 개발 v2 - 파트너 등록 통합, 역할 기반 권한 관리, 지도 기능 강화